Profs Agrawal and Daniel   *TA: Khaninder Kommagoni*

Assignment 2 – Computational Problem Solving

Primary objective: Develop familiarity with algorithms, data structures and problem solving

Secondary objective: Develop comfort with using GitHub and the IDE.

Estimated time per student: 20 hours

Introduction

Conversations with recent graduates suggests that employers expect most students to be comfortable with standard questions on problem solving, algorithms and data structures. Students have also shared that their prior experience often did not give them adequate introduction to algorithms, data structures and problem solving to be competitive in the market.

This assignment aims to give students the opportunity to familiarize themselves with problem solving, while simultaneously using basic sorting and searching algorithms to develop familiarity with algorithms and using simple data structures such as arrays, strings, dictionaries and hashsets to develop familiarity with the various data structures.

This is a group assignment, to give students the opportunity to learn to work in teams to develop deliverable solutions. To accomplish this, students are expected to use the GitHub source control and collaboration system to develop their solution.

Activity

All the methods are defined in Program.cs. Complete their definition to successfully run the program. To the extent possible, share the development effort. Teams with unequal development effort can lose credit.

You are free to modify the values of the arrays and variables in the main method.

NOTE 1: The method's declaration should not be modified under any circumstances, i.e., you cannot change the type of parameter or the return type.

Note 2: For full credit, your solution should meet the time and/ or space complexity constraints as well as any data structure/ algorithm recommendations specified.

Recommendation

Exam 1 is based heavily on the question types assigned in assignments 1 and 2. Therefore, though assignment 2 is a group assignment, it is in the best interests of every student to understand the logic and data structures underlying the solution to each question in this assignment.

Submission

Push the code to GitHub and submit the URL. Also, get the output from a sample run that shows the use of all required methods, take the screenshot and upload/push it to GitHub. Submit the self-reflection as a comment in Canvas.

Grading scheme

The first 8 questions are worth 1.2 points each. Question 9 is worth 1.4 points. Each question will be graded on the following aspects:

Logic (including appropriate organization of logic into methods)   : 1 (+0.2 for q9)

Handling all reasonable corner cases   : 0.2

In addition, the self-reflection for the overall assignment is worth 1 point. The self-reflection is to be included as a comment for the group. Please include the overall time taken to complete the assignment, learning from the assignment, and any recommendations for the assignment for future semesters.

Questions

Question 1:

Professor Manish receives an array of integer *points* sorted in ascending order, the task is to find the initial and final index of a given *target* point's value.

If the *target* point value is not found in the array of integers, return [-1,-1]

The professor had to leave for a conference at short notice and asked you to complete the task for him. He instructed you to limit the time complexity to O(n).

Example 1:

Input: marks = [5,6,6,9,9,12] target = 9

Output: [3,4]

Example 2:

Input: marks = [5,6,6,9,9,12] target = 10

Output: [-1,-1]

Note: The algorithm's runtime complexity must be O(n).

public int[] targetRange(int[] marks, int target)

Question 2:

Rocky is not aware of the inbuilt functions to split and reverse a string. He is given a string and he needs to reverse the order of characters in each word within a sentence while still preserving whitespace and initial word order. He is not allowed to use any predefined split and reverse function. He is requesting you to complete the method for him.

Example:

Input : "University of South Florida"

Output: "stisverinU fo htuoS adirolF"

Note: Time complexity should be O(n) where n is the length of string

Hint: Create your own split and reverse function.

Public static string StringReverse(string s)

Question 3:

Professor Clinton is given a sorted integer array. He needs to make the array elements distinct by increasing each value as needed, while minimizing the array sum. Professor Clinton thought this was an interesting exercise that the students might enjoy completing. Your job is to complete the method to print the minimum possible sum as output.

Example :

Input : arr[] = {2,2,3,5,6};   Output: 20

Explanation: Array is transformed to {2,3,4,5,6}. Sum becomes 2+3+4+5+6=20

Input arr[]= {40,40};   Output: 81

Explanation: Array is transformed to {40,41}. Sum becomes 81

Input: arr[] = {4,5,6,9};   Output : 24

Explanation: As the array is sorted and has distinct elements. So result is sum of each elements in the array.

Time Complexity: O(n)

public static int minSum(int[] arr)

Question 4:

You are given a string and your task is to sort the given string in decreasing order of frequency of occurrence of each character.

Example 1:

Input : "Dell"   Output: "llDe"

Explanation: "lleD" is also accepted. Frequency of l is 2 and other 2 characters is 1.

Example 2:

Input : eebhhh   Output: hhheeb

Example 3:

Input : yYkk   Output: kkYy

Explanation: "Yykk" is not a valid answer as Y and y are treated as two different characters.

Note: The solution must use a dictionary as the primary data structure.

public static string FreqSort(string s)

Question 5:

Rocky the Bull is new to programming and is having trouble understating the importance of time complexity. Professor Clinton assigned you the job of explaining time complexity to Rocky with the example below.

Given two arrays, write a function to compute their intersection.

Example 1:

Input: nums1 = [2,5,5,2], nums2 = [5,5]

Output: [5,5]

Example 2:

Input: nums1 = [3,6,2], nums2 = [6,3,6,7,3]

Output: [3,6]

Note:

- Each element in the result should appear as many times as it shows in both arrays.
- The result can be in any order.
- Submit 2 Solutions where time complexity of the methods is be **less than O(n^2)**.

Hint1: You can sort the both arrays and then compute the intersection, in this way you will achieve a time complexity of **O(n log n)**.

Hint2: If you make use of a dictionary, you need not sort the arrays to compute the intersection. The time complexity in this case will be **O(n)**.

public static int[] Intersect1(int[] nums1, int[] nums2)

public static int[] Intersect1(int[] nums1, int[] nums2)

Question 6:

You are given an array of characters and an integer k, and are required to find out whether there are two distinct indices i and j in the array such that arr[i]=arr[j] and the absolute difference between i and j is at most k.

Example 1:

Input: arr=[a,g,h,a], k=3

Output= true

Explanation= arr[0] and arr[3] contains "a"; hence the absolute difference between 0 and 3 is 3 which is equal to 3 therefore output is true.

Example 2: arr=[k,y,k,k], k=1

Output= true

Example 3:

Input: arr=[a,b,c,a,b,c], k=2

Output: false

Explanation= arr[0]=arr[3] but absolute difference is not at most 2

arr[1]=arr[4] but absolute difference is not at most 2

arr[2]=arr[5] but absolute difference is not at most 2

There fore output is false.

Note: The algorithm's runtime complexity must be in the order of O(n).

Hint : You can make use of a dictionary to obtain the required time complexity.

public bool ContainsDuplicate(char[] arr, int k)

Question 7:

Rocky, who lives in the virtual country of USF was gifted a gold rod of length n during the Week of Welcome. In this country, gold is always distributed in integer lengths, the price of a gold bar is equal to its length, i.e. a gold bar of length i has price i and the value of a collection of bars is equal to the product of the values of each bar. Rocky wants to maximize the value of this gift by breaking up the bar into appropriate lengths and using the money to create scholarships at USF. Your task is to help Rocky compute the optimal breakup for a gold bar of given integer length.

For Example, Consider a gold rod of length 4

Rods Combination	Profit
4	4
1,3	(1*3) = 3
2,2	(2*2) = 4
3,1	(3*1)=3
1,1,2	(1*1*2)=2
1,2,1	(1*2*1)=2
2,1,1	(2*1*1)=2
1,1,1,1	(1*1*1*1)=1

Best Solution: Cut the rod into 2 pieces of length 2 each such that the product is 2*2=4

Similarly for n=6 , (3*3) = 9

For n=15, (3*3*3*3*3)=243

Note: Use Recursion to solve the question

public static int GoldRod(int n)

Question 8:

Due to his huge fingers, our mascot, Rocky the Bull, is having a hard time typing on the keyboard. He wants you to create a method to fix some minor typos for him. You are given a list of words called userDict. Write a method which takes userDict and keyword as argument. The method should judge whether modifying **exactly** one character in the keyword into another character results in a word in the userDict.

Example 1:

userDict = ["rocky" , "usf" , "hello" , "apple"]

Input : "rocky" , Output: False

Explanation : As rocky already exists in userDict , any change in its characters would not satisfy our condition.

Input : "hhllo" , Output: True

Input : "appl" , Output: False

Input: "ucf" , Output: "usf"

Note:

1. You may assume that all characters are in lowercase letters a-z.

  public static bool DictSearch(string[] userDict, string keyword)

Question 9:

Summary: At a recent college reunion meeting, Prof. Agrawal's friend was wearing the t-shirt shown in the picture above. It was a gift from his niece. Appropriate assignment of numbers to each digit solves the puzzle above. In this question, write a general method to solve puzzles such as the above. The method should ask the user for the two input strings (e.g. uber, cool) and the output string (e.g. uncle) and identify a set of number assignments that solve the puzzle and print out the numbers.

Tip: There is no need to search for algorithms. It is fine to brute force all possible combinations. However, for full credit, use maximal organization of your code into the appropriate methods (e.g. a method to collect inputs, a method to identify unique characters in the strings, a method to test the currently assigned values etc).

returns   : nothing

return type : void

public static void solvePuzzle()

Appendix: Program.cs

```csharp
using System;

namespace Assignment2_CT_Spring2020
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Question 1");
            int[] l1 = new int[] { 5, 6, 6, 9, 9, 12 };
            int target = 9;
            int[] r = TargetRange(l1, target);
            // Write your code to print range r here

            Console.WriteLine("Question 2");
            string s = "University of South Florida";
            string rs = StringReverse(s);
            Console.WriteLine(rs);

            Console.WriteLine("Question 3");
            int[] l2 = new int[] { 2, 2, 3, 5, 6 };
            int sum = MinimumSum(l2);
            Console.WriteLine(sum);

            Console.WriteLine("Question 4");
            string s2 = "Dell";
            string sortedString = FreqSort(s2);
            Console.WriteLine(sortedString);

            Console.WriteLine("Question 5-Part 1");
            int[] nums1 = { 1, 2, 2, 1 };
            int[] nums2 = { 2, 2 };
            int[] intersect1 = Intersect1(nums1, nums2);
            Console.WriteLine("Part 1- Intersection of two arrays is: ");
            DisplayArray(intersect1);
            Console.WriteLine("\n");
            Console.WriteLine("Question 5-Part 2");
            int[] intersect2 = Intersect2(nums1, nums2);
            Console.WriteLine("Part 2- Intersection of two arrays is: ");
            DisplayArray(intersect2);
            Console.WriteLine("\n");

            Console.WriteLine("Question 6");
            char[] arr = new char[] { 'a', 'g', 'h', 'a' };
            int k = 3;
            Console.WriteLine(ContainsDuplicate(arr, k));

            Console.WriteLine("Question 7");
            int rodLength = 4;
            int priceProduct = GoldRod(rodLength);
            Console.WriteLine(priceProduct);

            Console.WriteLine("Question 8");
            string[] userDict = new string[] { "rocky", "usf", "hello", "apple" };
            string keyword = "hhllo";
```

```csharp
            Console.WriteLine(DictSearch(userDict, keyword));

            Console.WriteLine("Question 8");
            SolvePuzzle();

        }

        public static void DisplayArray(int[] a)
        {
            foreach (int n in a)
            {
                Console.Write(n + " ");
            }
        }

        public static int[] TargetRange(int[] l1, int t)
        {
            try
            {
                //Write your code here;
            }
            catch (Exception)
            {

                throw;
            }
            return new int[] { };
        }


        public static string StringReverse(string s)
        {
            try
            {
                //write your code here
            }
            catch (Exception)
            {

                throw;
            }
            return null;
        }
        public static int MinimumSum(int[] l2)
        {
            try
            {
                //Write your code here;
            }
            catch (Exception)
            {

                throw;
            }
            return 0;
        }

        public static string FreqSort(string s2)
```

```csharp
{
    try
    {
        //Write Your Code Here
    }
    catch (Exception)
    {

        throw;
    }
    return null;
}
public static int[] Intersect1(int[] nums1, int[] nums2)
{
    try
    {
        // Write your code here
    }
    catch
    {
        throw;
    }

    return new int[] { };
}

public static int[] Intersect2(int[] nums1, int[] nums2)
{
    try
    {
        // Write your code here
    }
    catch
    {
        throw;
    }

    return new int[] { };
}

public static bool ContainsDuplicate(char[] arr, int k)
{
    try
    {
        //Write your code here;
    }
    catch (Exception)
    {

        throw;
    }
    return default;
}
public static int GoldRod(int rodLength)
{
    try
    {
        //Write Your Code Here
```

```csharp
            }
            catch (Exception)
            {

                throw;
            }
            return 0;
        }
        public static bool DictSearch(string[] userDict, string keyword)
        {
            try
            {
                //Write Your Code Here
            }
            catch (Exception)
            {

                throw;
            }
            return default;
        }

        public static void SolvePuzzle()
        {
            try
            {
                //Write Your Code Here
            }
            catch (Exception)
            {

                throw;
            }
        }

    }
}
```